DOCUMENT RESUME

ED 328 213 IR 014 807

AUTHOR Elsom-Cook, Mark
TITLE Artificial Intelligence and Computer Assisted Instruction. CITE Report No. 4.
INSTITUTION Open Univ., Walton, Bletchley, Bucks (England). Inst. of Educational Technology.
PUB DATE 86
NOTE 15p.; A product of the Centre for Information Technology in Education (CITE). For related reports, see IR 014 806-809 and IR 014 811-813.
PUB TYPE Information Analyses (070) -- Viewpoints (120)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Artificial Intelligence; *Computer Assisted Instruction; Educational Environment; Expert Systems; *Instructional Systems; Interaction; *Programed Tutoring; Teacher Student Relationship

ABSTRACT

The purpose of the paper is to outline some of the major ways in which artificial intelligence research and techniques can affect usage of computers in an educational environment. The role of artificial intelligence is defined, and the difference between Computer Aided Instruction (CAI) and Intelligent Computer Aided Instruction (ICAI) is discussed. It is noted that the components of ICAI are similar to those of any traditional educational exchange in that they consist of a student, a teacher, and an environment. Distinctions are drawn between different existing programmed tutoring systems such as LOGO, GUIDON, Lisp, WHY, WEST, and IMPART in terms of: (1) user model, which forms a basis for assessment of the student, and constrains presentation of material, tasks, and generated examples to a form appropriate for the student; (2) domain knowledge, including domains with unconnected facts, closed domains with an underlying structure and a finite number of components, and open domains containing an infinite number of components and relationships; and (3) interaction style, which describes the relationship between the teacher and the student as well as the interaction between the environment and the student. It is concluded that further research on intelligent tutoring systems is required, but that one possible application of artificial intelligence technologies in education is the simplification of existing CAI approaches so that CAI can be used more extensively. (11 references) (DB)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

E.S. Henderson

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Artificial Intelligence and Computer Assisted Instruction.

Mark Elsom-Cook
Information technology centre,
Institute of Educational technology,
Open University,
Waltom Hall,
Milton Keynes,
MK7 6AA,
U.K.

1.Introduction

This paper is aimed at readers with some knowledge of traditional CAI systems, but with little or no knowledge of Artificial Intelligence. The purpose of the paper is to outline some of the major ways in which artificial intelligence research and techniques can affect our usage of computers in an educational environment.

2. Roles for Artificial Intelligence in Education.

Artificial Intelligence research has changed the nature of cognitive psychology, and is affecting many areas which are dependent upon psychological theories. In particular, it can cause us to reassess our approaches to teaching and learning. AI is capable of causing a major change in the way in which we think about the process of education and in the nature of educational activities. I wish to suggest that there are three major roles which Artificial Intelligence can play in education.

2.1) Discovery environments.

The role of a computer in Artificial intelligence research is to provide a means of formalizing theories (as a program) and a mechanism for exploring those theories. A computer program provides a concrete representation of a theory, and the behaviour of the program provides feedback on the consistency of that theory and a way to generate predictions associated with the theory. The claim is that this externalization of the theory facilitates correction of the model: debugging the program aids debugging the theory.

LOGO [Papert 1981] is an educational environment based upon these ideas for aiding AI research. The underlying assumption is that by providing a suitable formal framework (such as the programming language LOGO) a pupil can externalize their theories about an area and debug them, much as AI researchers do. This is based upon a Piagetian approach to learning which regards children as active constructors of theories about the world. Within LOGO, microworlds have been constructed to facilitate learning in the areas of geometry, music and language. LOGO is the best known and most successful attempt at such a system, but the ideas are more general than one particular programming language. By providing manipulable formal representations of a particular domain, a student can gain concrete experience of a domain which she may not be able to explore in reality (such as using engineering simulations to explore when a building will colllapse). This is an extension from the simple simulation programs which are often used in CAI, but which do not provide the student with the freedom and the facility to control the simulation.

2.2) Education theory.

An educational theory describes the goals of a particular educational system, provides some methods for achieving those goals, and (hopefully) includes some underlying model which explains why those methods will achieve those goals. These theories can be divided into two major components; a philosophical one and a psychological one [Peters 1973]. Artificial Intelligence has caused major changes in cognitive psychology and, since education theory is one

domain which is very heavily dependent upon cognitive psychology, there should be corresponding changes in the nature of education theory and educational research.

As in psychology, we can begin to formalize theories about educational methods in computer programs. These provide consistent and easily testable representations of the theories. As a side effect, we would expect that such programs can actually be evaluated by using them as teaching devices. The formal nature of programming means that we can begin to build detailed models of the nature of teachers and learners, and of the interaction between such systems.

2.3) Intelligent tutoring systems.

Intelligent tutoring systems are programs which are devised to act as teaching devices. They tackle the same problems with which a teacher in one-to-one interaction with a pupil would be faced. They are related to the use of machines to describe educational theories, since a good tutoring system is likely to embody some particular model of the teaching and learning process (though at present this is not always the case).

In the remainder of this paper we will focus on Intelligent Computer Aided Instruction (ICAI), which is the application of Intelligent tutoring systems in education.

3. The aims and features of ICAI.

Some of the advantages of ICAI are the same as for other uses of computers in an educational setting, however in many ways the use of Intelligent systems is drastically different from using non-intelligent machines. Before looking at the details of ICAI system, we will briefly summarize the aims of ICAI and the advantages that it offers to the pupil and teacher.

The overall goal of ICAI is to provide a computer-based teaching medium which offers the pupil a level and quality of educational interaction at least as good as that achieved by one-to-one interaction with a human teacher. This interaction should be highly adaptable to cope with individual differences, should be enjoyable and efficient as an educational mechanism, and should allow the pupil to control the educational experience whenever appropriate.

3.1) Advantages for the pupil.

The pupil is able to have a one to one educational interaction. This enables the pupil to progress at her own pace without worrying about the effect on fellow students.

The educational course will be adapted to the needs and abilities of the individual. Most teachers have neither the time nor the patience to provide the optimally adapted, supportive teaching for an individual student. As ICAI develops more powerful techniques for building models of pupils it becomes possible to provide levels of adaption much more detailed than the simple "prototyping" approach taken in CAI systems.

The machine can be available to the pupil at all times. This is very important when it is borne in mind that educational experiences rely heavily on the motivation of the learner, and that this motivation is best maintained by providing relevant and useful tuition at the time that the learner's interest is highest.

It has also been noted [Malone 1980] that computer use is intrinsically motivating. Pupils wish to interact with such machines, and the richer the behaviour which the machine can exhibit, the more they will be motivated towards such interactions. Intelligent machines have a fascination which can be of great value in maintaining the interest of pupils.

Some pupils are afraid of their teacher. By providing a machine with which to interact, the pupil can avoid this source of fear and gain confidence through interaction with the machine, which provides the same instant feedback that a teacher would provide, without being intimidated. There appears to be a strong case for children with severe social interaction problems learning to

overcome these barriers through interaction with a computer [Weir 1976].

Human teachers are constrained to one, or a small number, of educational styles which they have learnt, and these styles may not be appropriate for all pupils. An Intelligent computer can teach using an effectively infinite variety of methods (if they can be encoded suitably) and hence may be better for some pupils than some human teachers.

3.2) To the teacher or courseware designer?

Designing a piece of CAL courseware requires the author to structure the domain in a somewhat unnatural branching model. Material must be divided into chunks of approximately equal size, which are then connected into a structure by selection questions and problems. The courseware designer must attempt to envisage all possible situations which could arise, and to produce a segment of the courseware to deal with that situation. The courseware designer must learn to create a mapping from her own model of the domain into this counter-intuitive branching structure.

A key feature of an ICAI system is that this problem is overcome. The structure of the domain specific knowledge is explicitly represented, rather than being implicit in the whole program. This makes such structures closer to the cognitive structures of real teachers, and allows direct manipulation of those structures. It also follows from this that ICAI programs are much easier to modify, since it is possible to isolate a change and its consequences. Separating off the models of teaching and learning from the domain effectively reduces the preparation time for a teaching program in a new domain, because it is not necessary to rewrite this information for each new domain.

Because it can directly manipulate the domain representation, an ICAI system can have an "awareness" of its own limitations which does not exist in traditional CAI systems. This facilitates graceful degradation of the system in situations which are beyond its scope.

4. What is ICAI?

We will outline the main features which distinguish ICAI systems from other educational uses of computers, and then discuss each of the major components of such a system in more detail.

Generative.

Rather than providing the teaching system with a complete, explicitly enumerated set of behaviours to perform (such as questions to ask, advice to give, explicit tests and jumps to remedial material), Intelligent tutoring systems act at a higher level of abstraction than this. The system is provided with models of the domain, of methods for presenting material, methods for generating problems, and methods for selecting an action based on the response of the pupil to some action of the tutor. These techniques in combination produce a generative system, which applies these methods during a particular interaction to create new presentation sequences etc. in real time. Such generative systems can behave appropriately in situations which were not explicitly envisaged by their designers.

Adaptive.

The generative nature of the interaction permits the content, presentation method etc. to be changed during the course of the interaction. One use for such a facility is to allow the system to adapt the interaction to the preferred style for a particular user. This need not be a once-and-for-all change (e.g use teaching method 6 for student 7), but can be dynamically varying during the course of the interaction. A prerequisite for such an adaptive interaction is that the system should have facilities available for dynamically creating and modifying a model of the pupil. The sort of information to be kept in that model is discussed further below. Adaptiveness can take a variety of forms: Information presented, problems set etc. can be individualized, and interaction structures

and teaching methodology can be varied.

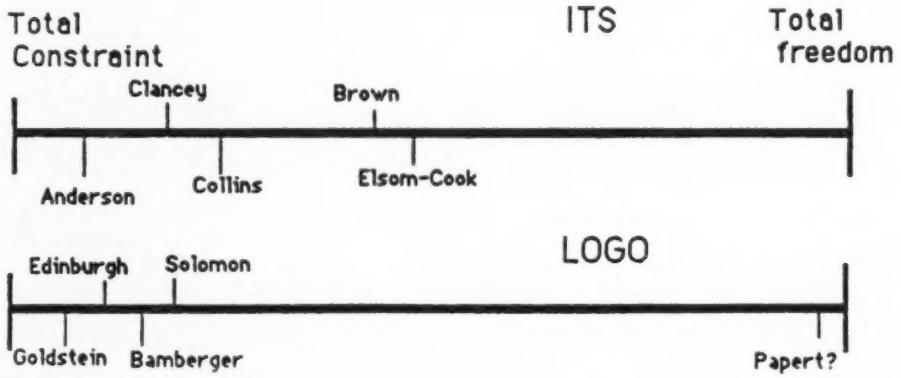
The range of tutoring systems.

Having outlined the central features of ICAI systems, it should be pointed out that there ar great variety of programs which come under this heading, with a wide variety of aims and capabilities. Not all the systems contain all the components which I am suggesting are essential: this is primarily because these systems are normally developed as research tools rather than complete tutoring systems suitable for actual use.

One important distinction to make when considering practical applications of such systems between stand-alone and supported use. Systems which are intended to operate autonomously a still not feasible as practical tools, but systems designed for use with teacher support are able to make contributions in the classroom. An example of one of the latter type of system is Buggy [Brown 1978]. This system diagnoses bugs in the subtraction of individual children, but has nc inbuilt capacity for remediation. Instead, the analyses are passed on to a human teacher who decides on the appropriate behaviour.

The choice of domain (which will be further discussed below) is also very important from the practical viewpoint. Small domains, where the structures to be taught are completely enumerable, provide considerably better bases for practical tutoring tools than do complex domains, which are often completely open-ended in the sort of problem-solving skills which they require.

Systems have differing forms of interaction with the student, and this is a major mechanism for categorising the educational form of the tutor. Some systems simply react to the input of the student, others act as instructors, rigidly defining the problems and activities which the pupil may perform, and there are a range of systems which lie between these two extremes, offering various degrees of guidance in teaching style. One feature of interaction which does not currently exist, which should be developed, is to permit a single tutoring system to support a variety of interactic styles. I wish to suggest that there are two major features to consider when exploring the nature the teaching interaction: one is the degree of planning ahead which occurs (many systems only ac on single utterances, but some maintain a plan built over many interaction steps), and the degree freedom/control which the tutor permits the pupil to have.



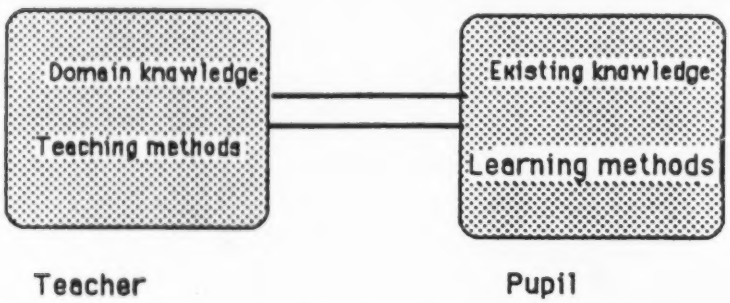
The diagram shows a number of existing tutoring systems (and, incidentally, a number of different approaches to the use of LOGO), spread along an axis intended to indicate the amount c freedom available to the pupil when using that system. Anderson's Lisp tutor and Clancey's GUIDON system impose strong constraints. Collins WHY system provides better support for interactions based on user-initiative, while Brown's WEST and Elsom-Cook's IMPART are gui

discovery systems (coaches) intended to provide more freedom. It is worth noting that all these systems lie on the more constrained part of the diagram. This is because it is easier to build a system that deals with a simple input than one which supports the full range of possible user-initiated interactions. There is considerable scope for development in this area.

4.1) What are the essential components of ICAI?

At the highest level, let us consider an educational exchange to be an interaction between three major components: a pupil, a teacher, and an environment. The overall goal of the interaction is that the pupil should build some model of the environment which is suitably integrated with her pre-existing knowledge structures and is in an accessible and usable form. In many interactions one of these components will be dominant and the others will be diminished or absent (for example, the environment is absent in "chalk and talk" approaches, the teacher is absent in genuine "free discovery learning" methods).

If we consider the active agents in the interaction (the pupil and teacher), then they have different cognitive structures which must be shared for the interaction to be successful. I would describe a beneficial educational interaction as one in which the exchange of knowledge is achieved through a basis of negotiation about these knowledge structures. The teacher comes to the interaction with a well-constructed (if idiosyncratic) model of the domain and a set of teaching methods which she can perform, while the pupil brings some pre-existing knowledge structures and some methods for learning.



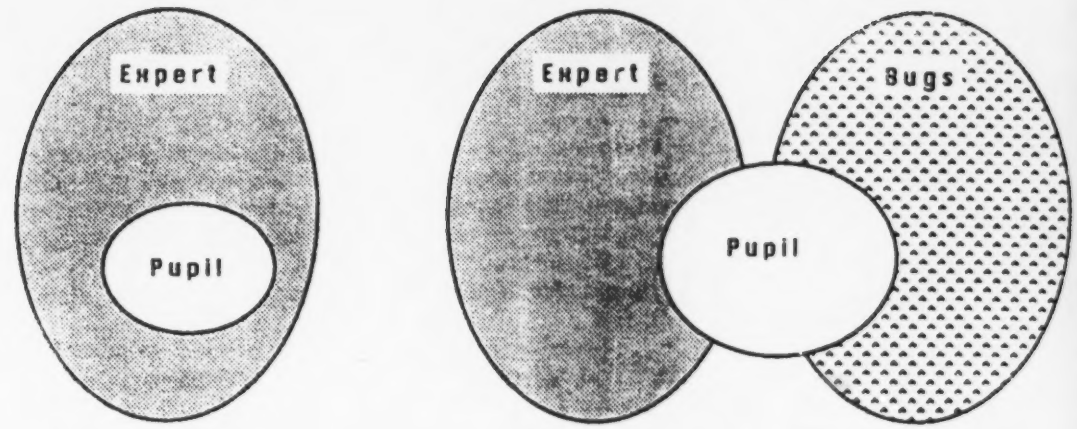
It is important to have such a model of the teaching interaction, because the nature of each component in an Intelligent tutoring system is determined by the purpose it is intended to fulfill. This purpose can only be determined if such a model exists (though it is often implicit in existing tutoring systems). We will now examine each of the major components of such a system.

User model.

The user model is the central component upon which the adaptive nature of the system lies. Its overall goal is to maintain a representation of the current state of the pupil with respect to the knowledge which the system is trying to teach (and possibly other things in addition). This model not only forms a basis for assessment of the pupil, but can also be used to constrain presentation of material, content of tutor-initiated tasks, nature of generated examples etc. to a form which is appropriate for the pupil at a particular time. The user-model can indicate which material has been learnt, which is to be learnt, and which is on a suitable "threshold of learnability".

There are many different forms of user-model in existing tutoring systems, from very simple (and often surprisingly efffective) ones, to models which use the full power of AI representation methods and inference techniques. The simplest models are simple numerical ones, which assign "goodness of knowledge" ratings to various identified aspects of a topic. This is the approach used in WEST [Burton 1982], which has a set of "issues" for which it attempts to recognise applications and misapplications in the actions of the pupil. If an issue is correctly applied, the priority for

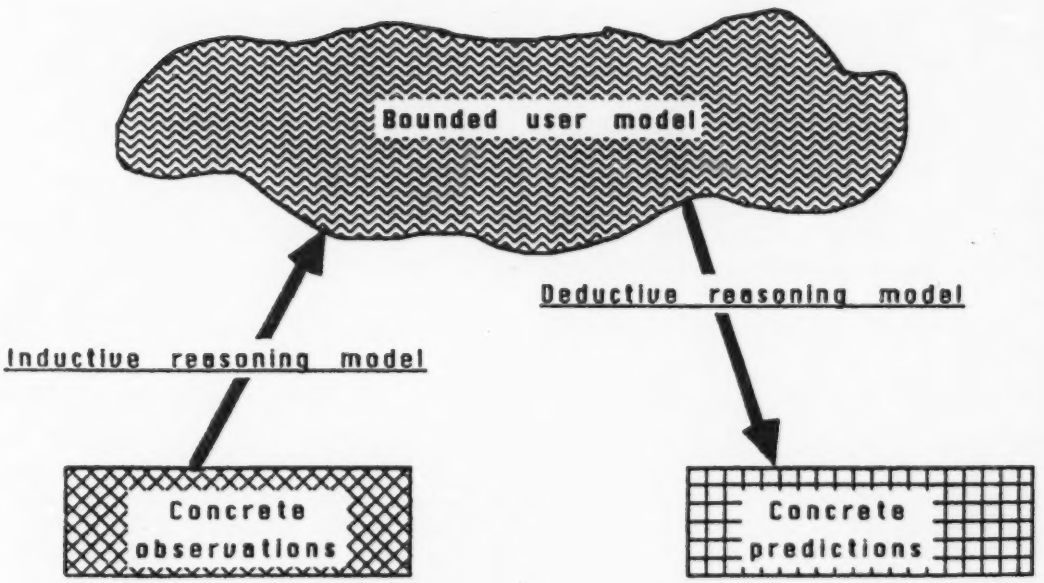
User modelling.



Subset

Perturbation

Closely linked to domain representation.



Closely linked to theories of learning.

tutoring directly about it is reduced, whereas if it is misapplied, the priority is increased.

More complex user models have a closer link with the way in which the system chooses to represent the domain knowledge. A common approach (which appears in a number of systems under several different names) is shown in the upper part of the diagram. The domain is represented as a number of small skills which are considered to be either present or absent (i.e. with no relevant internal structure). An expert in the domain is regarded as a set of such skills, while a pupil is considered to possess only a subset of the necessary skills. The goal of the tutoring system is to increase the number of skills which the pupil has until they include all those of the expert. This can be described as a **subset** modelling method. This method relies on a somewhat simplified view of teaching, in which the goal is to make the pupils into little copies of the teacher, with no concern for integrating this knowledge with their pre-existing knowledge structures: the pupil is viewed as a bucket into which knowledge is to be piled.

The second diagram shows an expansion of this approach. In this case, it is acknowledged that the pupil may (due to factors beyond the control of the system) misleam the skills of the domain. This is dealt with by adding a second set of skills (called **bugs** or **malrules**) which are not possessed by the expert, but which correspond to errorful versions of correct skills. Buggy [Brown 1978] and Proust [Spohrer 1985] both rely on large databases of bugs collected in empirical surveys. This approach represents an improvement on the first in the sense that an "expert on errors" is incorporated into the system, but the goal is still to make the pupil into a little copy of the teacher.

The approach to user modelling which I wish to advocate here is depicted in the lower part of the diagram. Instead of attempting to build an exact model of what the pupil knows, it is based around the idea of putting upper and lower bounds on that knowledge, and attempting to gradually draw those bounds together. As is shown in the diagram, this is achieved by using a model of the learning process to infer these bounds from the same concrete observations which the pupil can make. These bounds are then regarded as hypotheses which can be tested by using them to make predictions about the concrete behaviour of the pupil. This approach relies on building models of inductive and deductive reasoning into the system, and such an approach will not realise its full potential until our models of these processes are considerably better developed than they are at present. However, it is possible to provide remarkably powerful tutoring behaviour using simple models of these processes (provided the domain in chosen carefully): IMPART [Elsom-Cook 1984] uses a naive machine-learning model of inductive reasoning with remarkable effectiveness in the domain of tutoring programming.

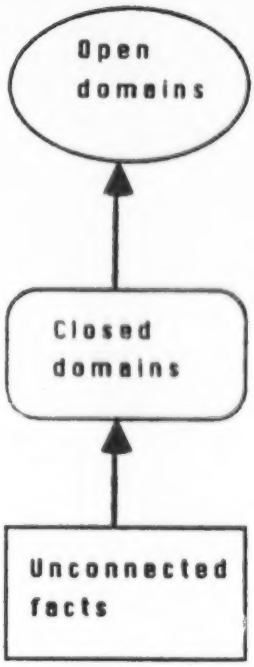
Domain knowledge.

The full range of knowledge representation techniques of Artificial Intelligence have been applied in the tutoring domain. The way in which this knowledge is represented is important, since it may be used for a number of different activities: problem-generation, problem-solving and information production all rely on this source of knowledge. Each of these activities will be carried out by a separate program acting on the same knowledge structure.

The diagram suggests a three-level taxonomy for the sorts of domains which one may attempt to tutor. At the simplest level are domains composed of unconnected facts which must be acquired by rote learning. These domains do not require links to existing knowledge structures, nor special concerns about methods for using the information. Such domains are, however, rare. Even apparently unconnected domains, such as simple geography (as taught by SCHOLAR [Carbonell 1970]) have hidden structure. An unlikely example of such a domain is the rule-base used in an expert-system: tutoring systems which attempt to use expert systems as a base are selecting a simple problem in this respect.

The second level of the diagram represents closed domains. These are domains which possess structure, but where the components of the domain and their relationships can, in some sense, be enumerated. Methods of using the knowledge must also be taught, but again, there are only a finite number of such methods available. These domains are best suited to the expert-based modelling methods described earlier.

Domain knowledge



Expert based user modelling fails.

(e.g. programming)

Enumerable - could be lookup.

(e.g. rainfall (WHY))

Rote learning.

(e.g. derby winners, geography, GUIDON!)

How you represent the domain is closely related to what you intend to do with the representation.

In fact:

semantic nets: Scholar, Why, Trill

Expert systems: Guidon

Production rules: Buggy, O'Shea, Anderson, West?

Wierdos: Sophie

The highest level of the diagram describes open domains. These are domains in which not everything to be taught is explicitly enumerable and simple user-modelling techniques fail. An example of such a domain is teaching programming, where any given problem may have an infinite number of "correct" solutions. No existing tutoring systems provide powerful aid in this type of domain.

If we examine the range of knowledge representation techniques used in tutoring systems we find that there are remarkably few. Most systems use either semantic nets or production rules. Production rules are well suited to expert-based user-modelling, because each rule can correspond to a primitive skill or bug, and the user-model is simply a collection of these rules drawn from a larger knowledge base. The only attempt to use an expert system directly as a knowledge representation is GUIDON[Clancey 1979], which is built upon the medical expert system MYCIN. This system demonstrated that the form of knowledge built into an expert system is not an appropriate basis for tutoring (though if the database were originally constructed from an educational viewpoint this would not necessarily be the case).

In terms of the link between educational research and tutoring systems, the knowledge representation issue runs parallel to questions of curriculum design. Unfortunately, these fields have not yet been adequately linked together.

Interaction style.

It has already been noted that the style of interaction is very important in deciding the educational form of the tutor. The amount of freedom which pupils have remains an unresolved and difficult issue in education, and the emphasis on one-to-one tutoring in ICAI further highlights this difficulty. Most systems use a highly constrained interaction and, if the pupil is given any choice at all, it is in the form of selecting one of a number of options. I wish to suggest that there are four main categories of tutoring system: reactive systems, mixed-initiative systems and goal-directed/planning systems.

Reactive systems are ones which deal with individual dialogue units independently. They process a single utterance by the pupil and react to it without reference to the context or to any larger structures in the dialogue. This does not imply that they are ceding control of the interaction to the pupil: the reaction may be telling the pupil what to do next. Most existing tutoring systems operate in this manner.

Mixed-initiative systems attempt to recognise responses in which the pupil is attempting to take control of the interaction, and allow that takeover to occur. They also look for opportunities to gain control of the interaction. These systems require some model of conversation structure, but in most implemented systems (e.g. SCHOLAR [Carbonell 1970]) this model is very simple.

Goal-directed, planning interaction mechanisms are currently very rare in Intelligent tutoring systems. These systems maintain a complete goal-structure for the dialogue and attempt to satisfy tutoring goals through that interaction. They have a higher level of "awareness" of what they are saying, and the context in which it occurs. This is based upon research towards AI models of conversation, such as that of Levin and Moore [Levin 1980].

In the perfect tutoring system, the interaction knowledge would be separable from the rest of the system, and represented explicitly. This is necessary to facilitate the variety of interaction styles discussed earlier. In fact, this knowledge is "wired in" for almost all existing systems, so that it cannot be changed. Four interesting uses of dialogue are embodied in the systems WEST, WHY, GUIDON and IMPART.

WEST uses canned speeches and examples associated with particular issues in the domain which it is tutoring. When an utterance contains one of these issues, the appropriate canned text is presented. This is a purely reactive system with no structured knowledge about the dialogue.

WHY tutors by organising its knowledge according to a set of tutoring rules. These provide

a short-range structure spanning several utterances. The rules are determined from empirical observation of actual teachers.

GUIDON maintains a structured interaction in which the tutor has control. This makes use of the observation that dialogues about particular tasks often follow the structure of the task itself. In GUIDON the task is medical diagnosis. The underlying expert system builds a tree corresponding to the method of diagnosis, and the tutoring system follows the structure of this tree in teaching the student to perform a similar diagnosis.

IMPART uses a goal-directed approach in which functional goals ("dialogue games") and domain-related goals ("topic controllers") are combined with the aid of a planning mechanism. This generates (and modifies continuously) a complete plan for the interaction with the pupil. These goals are intended to be reversible, so that the system can respond to pupil utterances in terms of the pupil's goal structure.

Assessment mechanisms.

How to assess the progress or state of learning of a pupil is a difficult educational problem. In an Intelligent tutoring system, it is necessary to carry out such assessment continuously during the educational interaction in order to decide on the most appropriate next step. None of the existing tutoring systems provide anything more than a rudimentary means of assessment, and the problem is, in general, a difficult and untackled issue. In the case of expert-based (subset and perturbation) user models like those described above, there is a simple solution to the problem which is more or less forced on the system designer: the assessment of the pupil consists of the set of skills which the pupil has acquired (or the discrepancy between that set and the expert set).

Having discussed all the components of the system individually, it should be pointed out that it is not always the case that this separability can be achieved in practice. Many of the components can overlap in a practical implementation.

5. Practical possibilities for existing technologies.

The Intelligent Tutoring systems discussed here constitute an area which requires further research. It will be some time before such systems are ready for classroom use. However, I wish to suggest that many of the ideas from Intelligent tutoring systems can be used individually in simple forms to enhance current applications of computers in education. In this section some ideas for extensions to simple CAI programs will be outlined.

One possibility for the practical application of existing AI technologies in education is to simplify the production and maintenance of traditional CAI approaches such as author language usage. The intention here is that by reducing the time taken to generate courseware, a greater variety of courseware will be produced, enhancing the educational experience to be achieved using this medium. Within this domain there are several possible extensions:

1: Tools for thinking about material structure. Instead of forcing the courseware designer to restructure the topic material (unaided) in the linear branching form required for software, tools can be developed to enable the designer to specify the curriculum in a more abstract sense (using AI knowledge representation techniques). Programs can then be provided to automate (at least to some extent) the transformation of this material into a linear structure. For example, if the content of each frame can be specified in some machine-understandable semantic form then links between frames and possible branch points can be generated.

2: A formal specification of the frame content would similarly allow the system to be more flexible in choosing between sections of text. Slightly more sophisticated processing of user responses at branch points would permit a sufficiently detailed analysis to allow a general purpose algorithm to select the next frame from a large database of frames. Such an approach could be the default behaviour of an authoring system, with facilities for the course developer to override that behaviour when necessary.

3: Automatic generation of problems (in simple domains) is also a technologically feasible extension. Given a basic representation of the domain, some quite simple algorithms can be used to produce a range of "possible problems" from which the courseware designer can then choose.

4: User-modelling techniques in author languages are primitive and restrictive. Simply providing an AI-based user model can enable more sophisticated structuring decisions to be taken in designing the course.

5: Taking the idea of specifying a formal equivalent of the contents for each frame, one possible area for further investigation is the use of that formal specification for generating the content of the frame itself: this is a considerable step on the way to providing an adaptive interaction.

6: Ideas from research on Human-computer interaction, and more structured interaction styles could easily be incorporated into CAL software. This is becoming more feasible as user-interface design languages begin to appear.

All these techniques can be regarded as tools moving towards a goal of automatic generation of courseware segments. Other possibilities exist outside the range of existing CAI systems, by producing programs which only deal with part of the tutoring task.

Diagnosis programs, which aim to analyse the difficulties of a particular individual without tutoring or repairing them, are feasible in some domains. These systems avoid the problems of tutoring by performing an analysis of the pupil and producing a report which can be used by a human teacher. This is essentially a focus on the assessment aspects discussed earlier.

Advising programs assume that they are dealing with an expert in the domain which they are teaching. There is therefore no requirement to deal with the introduction of new concepts, or with questions such as curriculum structure. The adviser sits alongside an expert performing a task and attempts to build a model of the expert/task which is sufficient to permit it to contribute information from its own knowledge base.

References.

[Brown 1978] J.S.Brown,R.R.Burton - Diagnostic models for procedural bugs in basic mathematical skills
Cognitive science 2 1978

[Burton 1982] R.R.Burton,J.S.Brown - An investigation of computer coaching for informal learning activities
in D.Sleeman,J.S.Brown (eds) Intelligent tutoring systems. 1982

[Carbonell 1970] J.R.Carbonell - Mixed initiative man-computer instructional dialogues
BBN report no. 1971 1970

[Clancey 1979] W.J.Clancey - Dialogue management for rule-based tutorials.
in Proceedings of 6th IJCAI 1979

[Elsom-Cook 1984] M.T.Elsom-Cook - Design considerations of an Intelligent tutoring system for programming languages.
Ph.D thesis Warwick university (UK) 1984

[Levin 1980] J.A.Levin,J.A.Moore - Dialogue games: metacommunication structures for natural language interaction.
Cognitive science 11 1980

[Malone 1980] T.W.Malone - What makes things fun to learn? A study of intrinsically motivating computer games
XEROX-PARC report CIS-7 1980

[Papert 1981] S.Papert - Mindstorms
Harvester press 1980

[Peters 1973] R.S.Peters - Freedom and the development of the free man
in J.F.Doyle (ed) Educational Judjements 1973

[Spohrer 1985] J.C.Spohrer,E.Soloway,E.Pope - Where the bugs are.
in Proceedings of CHI '85 1985

[Weir 1976] S.Weir,R.Emanuel - Using LOGO to catalyse communication in an autistic child.
Edinburgh DAI report 15 1976

CENTRE FOR INFORMATION TECHNLOGY IN EDUCATION

List of CITE Reports

These reports may be obtained from:

Hansa Solanki, Institute of Educational Technology, The Open University, Walton Hall, MILTON KEYNES, MK7 6AA, England.

Report No.	Title and Author
1	A.T. Vincent, (1985) Computing and the Blind.
2	A. Jones, G. Kirkup, J. Morrison (1985) A Trial of Home Based Computer Terminals.
3	Gill Kirkup, (1985) The Present and Potential Use of Ceefax in the Open University.
4	Mark Elsom-Cook, (1986) Artificial Intelligence and Computer Assisted Instruction.